



MAUI HIGH PERFORMANCE
COMPUTING CENTER

**Support for DoD Challenge Project
“Unsteady Aerodynamics of Aircraft Maneuvering at High
Angles of Attack”
at the Maui High Performance Computing Center**

**DoD Users’ Group Conference
June 6, 2000**

**Dr. James S. Newhouse
Challenge Project Coordinator**

PRESENTATION OVERVIEW

- MHPCC BACKGROUND
- STRATEGY
- RESOURCES
- SUPPORT FOR CHALLENGE PROJECT C33
“Unsteady Aerodynamics of Aircraft Maneuvering
at High Angles of Attack”

MHPCC



Approximately 27,000 Square Feet
Conference and Training Rooms
Classified Processing
Network Communications
Visualization

Located in the Maui Research and Technology Park

MHPCC

Affiliations and Collaborations

HPCMP

Distributed Center of the DOD
High Performance Computing
Modernization Program

SuperNode of the NSF
National Computational Science Alliance



Strategic Center of the
University of New Mexico

Member of Hawaii State
Science and Technology Community



MHPCC

A DOD Distributed Center

**Deputy Under Secretary of Defense
(Science and Technology)**

High Performance Computing Modernization Program

Major Shared Resource Centers

- Aeronautical Systems Center (ASC)
- Army Research Laboratory Aberdeen Proving Ground (ARL-APG)
- Corps of Engineers Waterways Experiment Station (CEWES)
- Naval Oceanographic Office (NAVO)

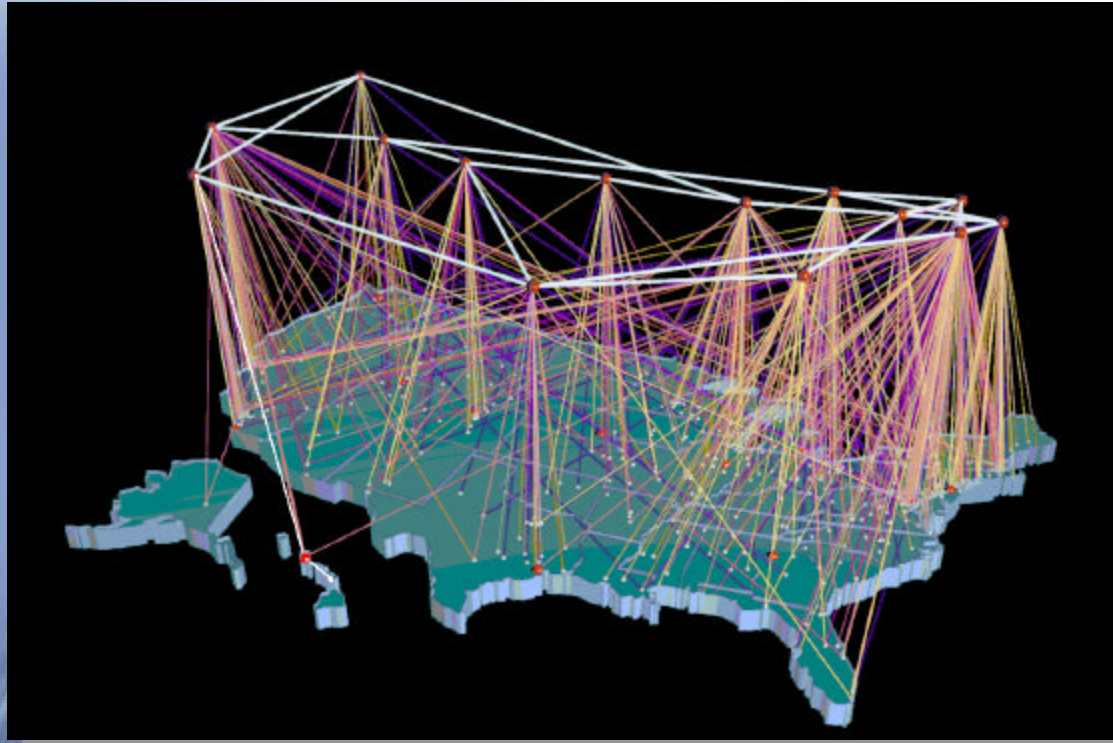


Distributed Centers

- Air Force Air Armament Center (AAC)
- Air Force Research Laboratory (AFRL/IF)
- Army High Performance Computing Research Center (AHPARC)
- Arnold Engineering Development Center (AEDC/SDC)
- Arctic Region Supercomputing Center (ARSC)
- Joint National Test Facility (JNTF)
- **Maui High Performance Computing Center (MHPCC)**
- Naval Air Warfare Center Aircraft Division (NAWC-AD)
- Naval Air Warfare Center Weapons Division (NAWC-WD)
- Naval Research Laboratory - DC (NRL-DC)
- Redstone Technical Test Center (RTTC)
- Space and Missile Defense Command (SMDC)
- Space and Naval Warfare Systems Center (SSCSD)
- Tank-Automotive Research, Development and Engineering Center (TARDEC)
- White Sands Missile Range (WSMR)

MHPCC

Internet Interconnectivity

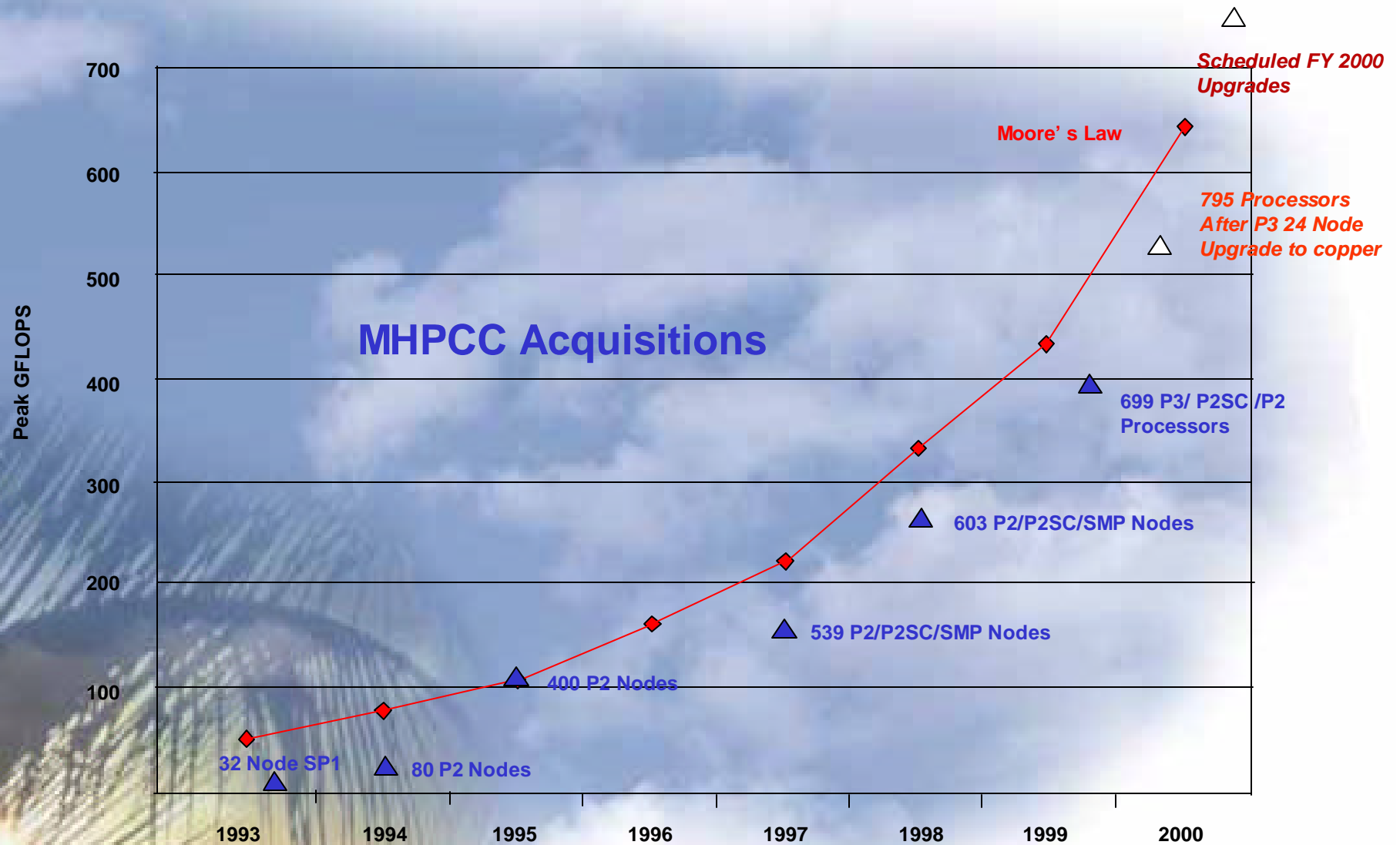


Defense Research and Engineering Network T3

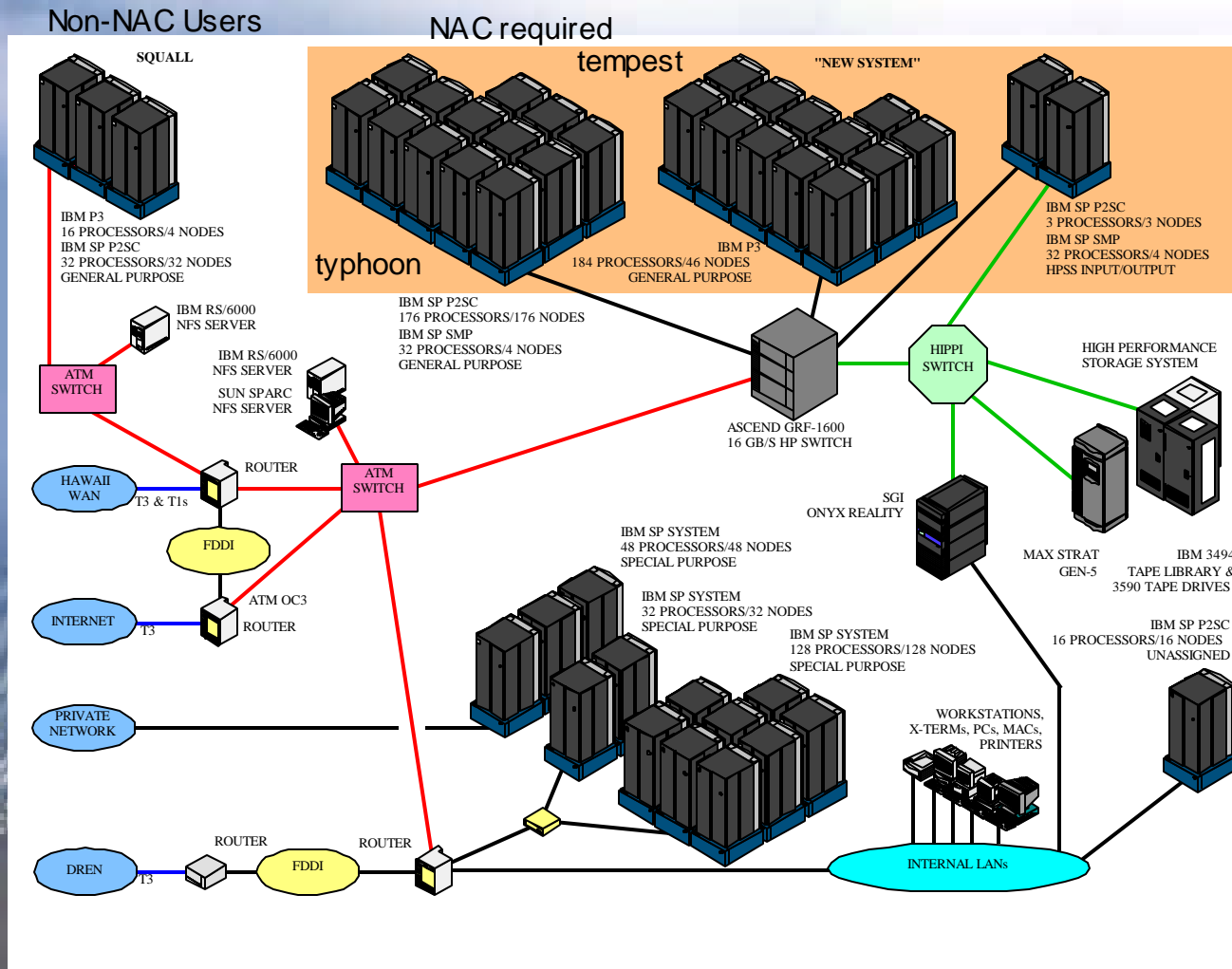
Second T3 Connection to the Internet Backbone

Hawaii Wide Area Network

The Pace of Advancement



Computing Resources as of January 2000



IBM SP SYSTEMS

- New System (USM)
 - 184 P3 processors
 - 179 P2SC processors
 - 64 SMP processors
- Squall (UNSM)
 - 32 P2SC processors
 - 16 P3 processors
- Collateral Secret
 - 80 P2 processors
- TS/SCI
 - 96 P2 processors
 - 32 P2SC processors
- Unassigned
 - 16 P2 processors

COMPUTING CAPABILITY

- 699 processors
- 388 GFLOPS

MEMORY

- 250 GB total memory

STORAGE

- 5.8 TB disk storage
- 20 TB on-line tape storage

Pending Upgrades



- **Current IBM Power 3 System (*tempest*)**
 - 24 Nodes (96 processors) to be upgraded to Nighthawk-2 (NH-2), copper technology in FY00
 - Each NH-2 processor rated at 1.5 GFlops
 - NH-1 processors that are being replaced will be retained
- **TS and Secret configurations will be replaced in FY00**
 - IBM Winterhawk-2 (WH-2) SMP technology
 - Each WH-2 processor rated at 1.5 GFlops
 - 80 nodes (320 processors) with an aggregate rating of 480 GFlops
- **IBM NH-2, 16-way, SMP systems to be added in FY01**
 - 16 - 24 nodes with an aggregate peak rating in excess of 1 TFlops
- **Large Linux supercluster to be deployed in FY00**
 - 128 - 512 total processors
 - Each processor rated at ~ 750+ Mflops

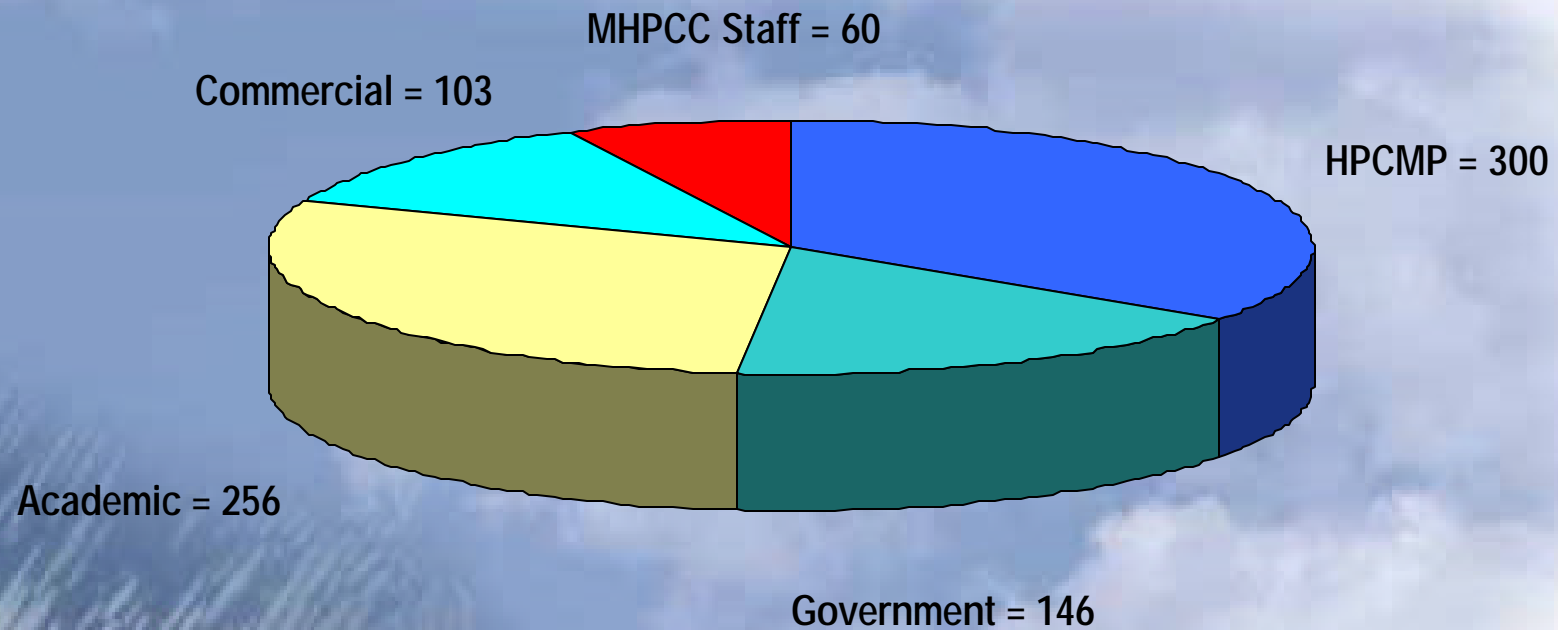
MHPCC Schedule of System Modifications May through December 2000

Anticipated System Upgrades Scheduled Dates

1. Delivery and installation of replacement processor boards for the 50 Nighthawk-1 (NH-1) nodes in tempest. These are the general availability processor boards for NH-1 systems. We are currently using pre-production processor boards. **June 15 – June 30, 2000**
2. Acquisition and installation of the \$3.4 million upgrade. This will include:
48-4 processor, Winterhawk-2 (WH-2), SMP nodes (5 frames) for the TS environment and 32-4 processor, Winterhawk-2 (WH-2), SMP nodes (2 frames) for the collateral Secret environment. **July-August 2000**
3. 24 Node NH-1 to NH-2 upgrade for tempest. 24 nodes will have their processor boards replaced with the NH-2 technology (1.5 Gflops per processor). The 24 processor boards that will be removed will be re-deployed in 24 of the remaining NH-1 nodes. **August-September 2000**
4. Colony Switch upgrade. Upgrade to the current switch for tempest. **August-September 2000**
5. Acquisition and deployment of a 128-256 processor Linux cluster. **August-September 2000**
6. Acquisition and implementation of 22 – 16 Processor, NightHawk-2 (NH-2), SMP nodes (6 frames) to be added to tempest. **October-November 2000**



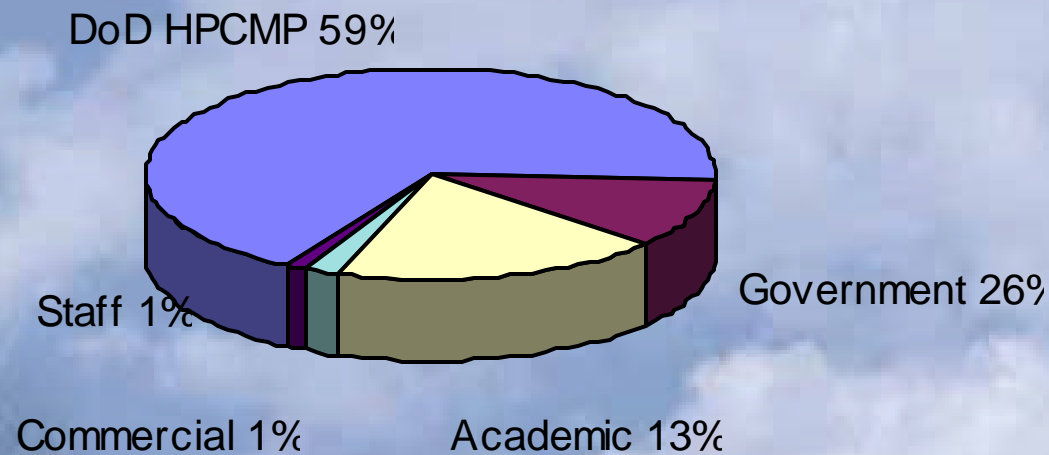
User Population



Total Number of Accounts = 865

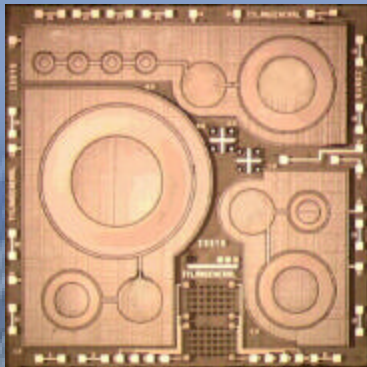
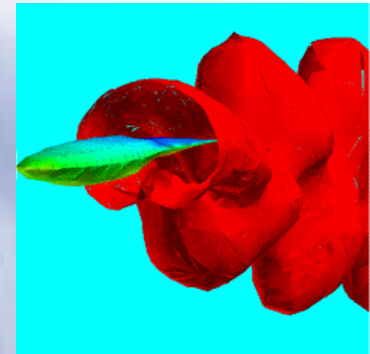
Node Hour Distribution on Unclassified Systems

FY 99 Usag



Navy Challenge Projects

Parallel Simulations of Flow-Structure Interactions
Coupled Dynamics of Flow-Structure Interactions
Using Direct Numerical Simulation



Atomistic Simulation of Micro Electro Mechanical Systems (MEMS) Devices via the Coupling of Length Scales
Dynamic and Temperature Dependent Behavior of
MEMS Devices with Their Atomic Constituents

Time-Domain Computational Ship Hydrodynamics
Microscale and Macroscale Hydrodynamic Motions
for the Design of the DDG-51

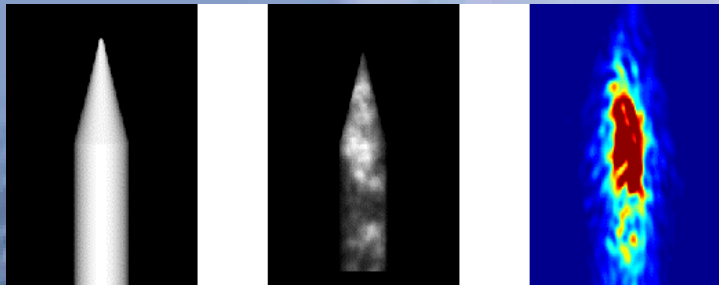
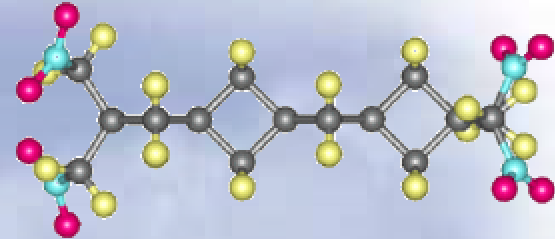


MHPCC

Air Force Challenge Projects

New Materials Design

High Energy Materials Critical for the Design of Next Generation Fuels



Airborne Laser Challenge Project

Effects of Optical Propagation on Optical Tracking and Adaptive Optics

Unsteady Aerodynamics of Aircraft Maneuvering at High Angles of Attack
Unsteady Aerodynamics Acting on Fighter Configurations at High Angles of Attack



Cobalt and the Power3

How does one enhance the performance of a code such as Cobalt, which has been given an excellent MPI treatment, on the new SMP architectures?

`MP_SHARED_MEMORY = YES; MP_WAIT_MODE = POLL`

Restructuring for OpenMP or a hybrid OpenMP/MPI seems like trying to fix a code that isn't broken and could destroy the almost perfect load balance that has been achieved in Cobalt (comment by Matt Grismer).

One can enhance a code that makes extensive use of collective communication calls in MPI with a package under development by John Hague at IBM called TurboMPI. TurboMPI is incomplete as yet, and eventually is targeted to become a standard part of the IBM's MPI for the SP.

“Because the entire pattern of communication between the processors is known for each call, there is excellent opportunity with these calls to make the best use of shared memory and distributed memory.”

Enhancement of Cobalt Performance on Power 3 Architecture

- Excellent MPI Program
- No plans for OpenMP or Hybrid OpenMP/MPI Treatment
- TurboMPI allows improvement of Collective Communications
 - Currently Limited to MPI_BARRIER, MPI_ALLREDUCE, MPI_REDUCE, MPI_BCAST, MPI_ALLTOALL
 - MPI_GATHER, others will be added (find use in many applications such as Full Configuration Interaction (Comp Chem--my field))
- Treatment Remains MPI
- MPI_ALLREDUCE, MPI_REDUCE, and MPI_BCAST are used in Cobalt

Normal Cobalt Run

F18-E/F Abrupt Wing Stall - AOA = 6 SST - 23 May 2000

5,312,398 Cells

64 processors

Clock Run Time: 4522.91 seconds

Solution Time: 3655.09 seconds

Overhead Time: 867.81 seconds

CPU Rate: 73.1 seconds/iteration
13.8 microseconds/cell/iteration

Collective Communications Calls in Cobalt

MPI_ALLREDUCE (on the order of 10 calls altogether in Cobalt)

Call to MPI_ALLREDUCE (in solver.f of Cobalt)

```
call MPI_ALLREDUCE(DTMINZ,DTMIN,1,MPI_FP,MPI_MIN,MPI_COMM_WORLD,IERR)
```

MPI_REDUCE (on the order of 30 calls altogether in Cobalt)

Call to MPI_REDUCE (in perforc.f of Cobalt)

```
call MPI_REDUCE(PERFM,TPERFM,6,MPI_FP,MPI_SUM,0,MPI_COMM_WORLD,IERR)
```

MPI_BCAST (on the order of 100 calls altogether in Cobalt)

Call to MPI_BCAST (in walldst.f of Cobalt)

```
call MPI_BCAST(XCTMP,int(ZCELLS(NZ)*3),MPI_FP,int(NZ-1),MPI_COMM_WORLD,IERR)
```

TurboMPI

John Hague

ACTC IBM

“..created to enhance the performance of IBM's standard MPI for the following collective communication calls:

MPI_BARRIER

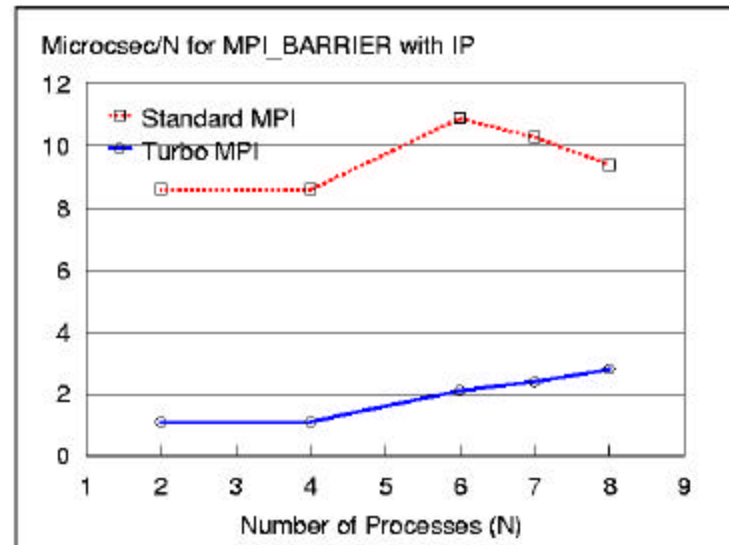
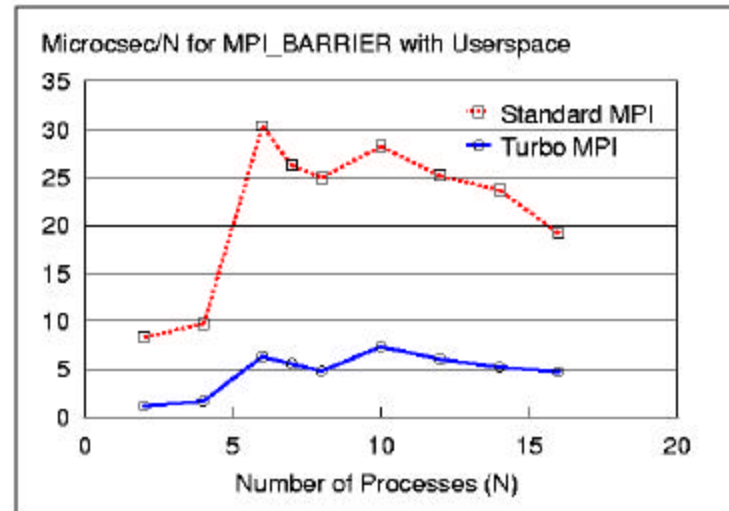
MPI_ALLREDUCE


MPI_REDUCE

MPI_BCAST

MPI_ALLTOALL”

MHPCC



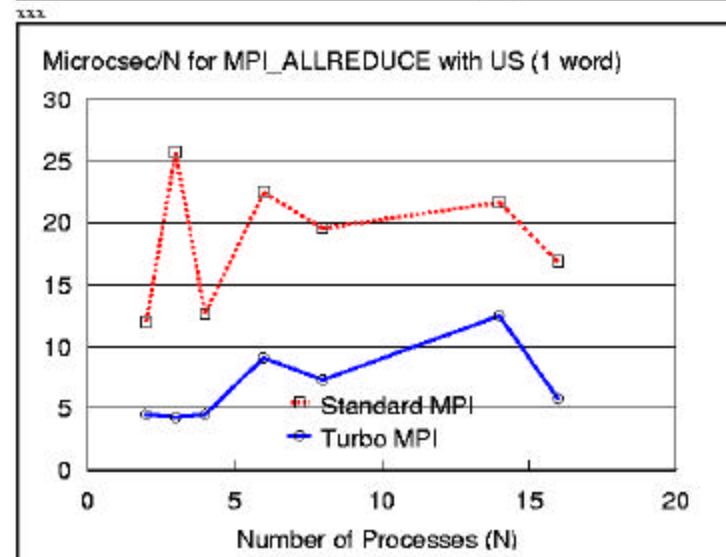
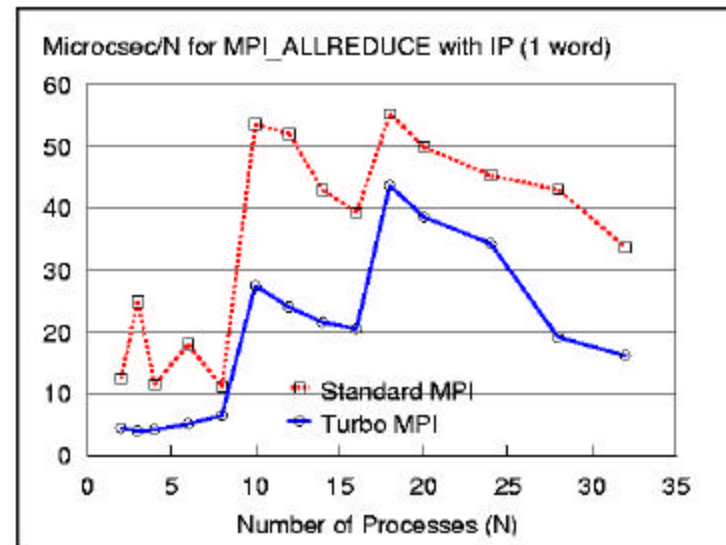


“Because the entire pattern of communication between the processors is known for each call, there is excellent opportunity with these calls to make the best use of shared memory and distributed memory.”

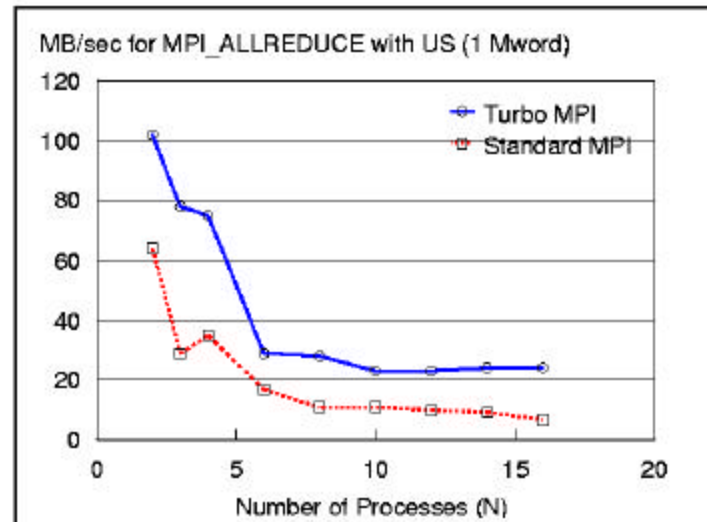
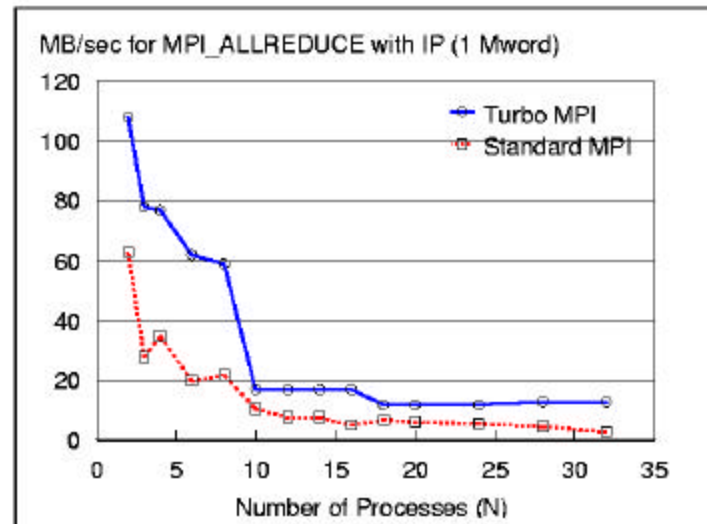
“Because the entire pattern of communication between the processors is known for each call, there is excellent opportunity with these calls to make the best use of shared memory and distributed memory.”

MPI_ALLREDUCE (MIN, MAX, and SUM)

MHPCC



MPI_ALLREDUCE (MIN, MAX, and SUM)



MPI_REDUCE (MIN, MAX, and SUM)

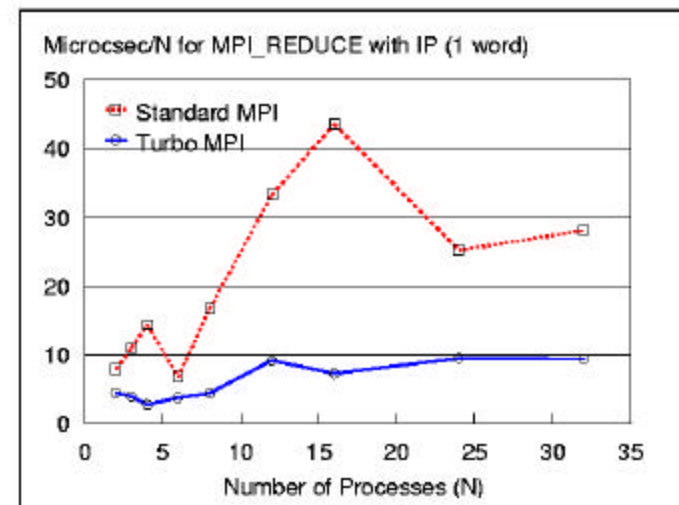
MPI_REDUCE

MPI_REDUCE is currently enabled for MIN, MAX, and SUM. The other standard operations are BRCD, LAND, LOR, LXOR, BAND, BOR, BXOR, MINLOC, and MAXLOC. If any of these operations are used then a single message will be printed indicating that standard MPI will be used.

The time taken for MPI_REDUCE to sum one double precision word across all processors was measured for 2 to 32 processes on four 8-processor Power3 high nodes. The times are shown as Microsec/N (where N is the number of processes), simply to aid representation on a single chart.

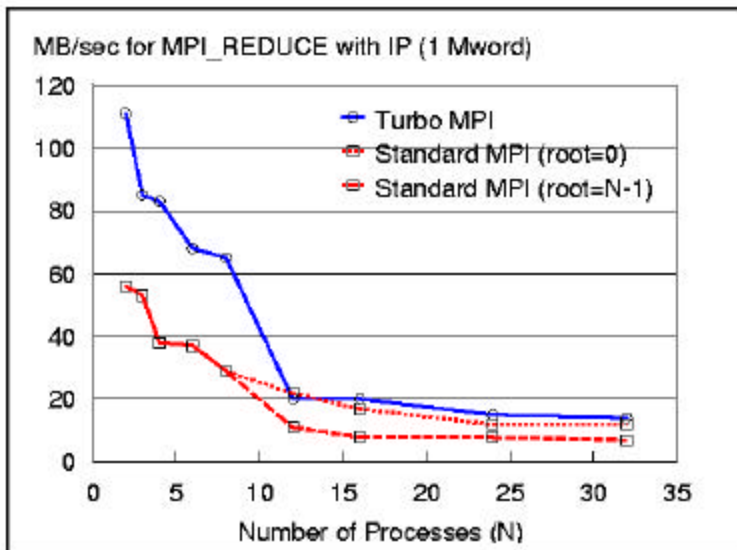
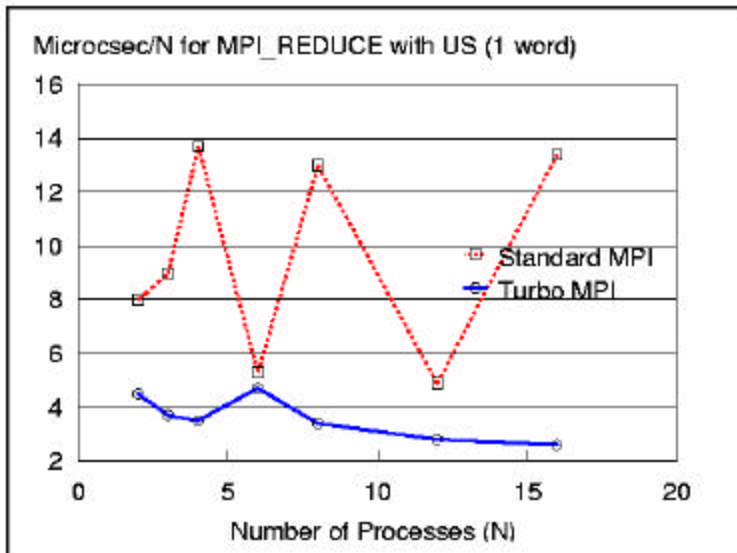
The IP communication was made using the SP switch connection between the nodes.

The summation rate for large arrays was measured in MB/sec for the summation across all processors of an array of one million double precision words. For standard MPI, the rate depended on the value of the root processor.

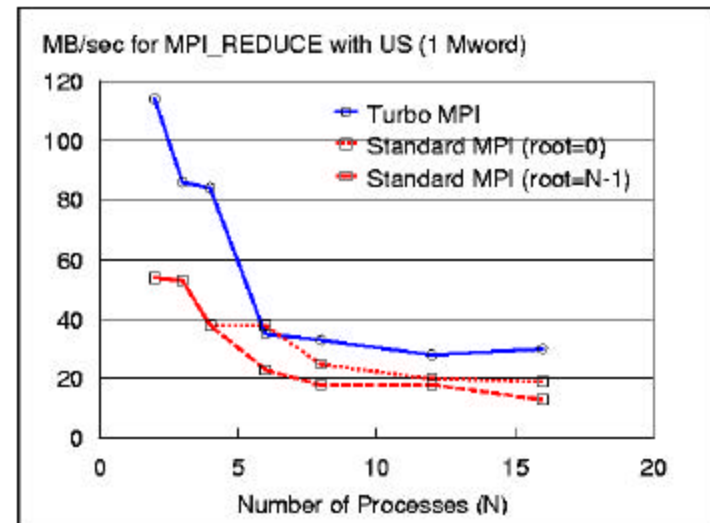


XX

MPI_REDUCE (MIN, MAX, and SUM)



MPI_REDUCE (MIN, MAX, and SUM)



XXX

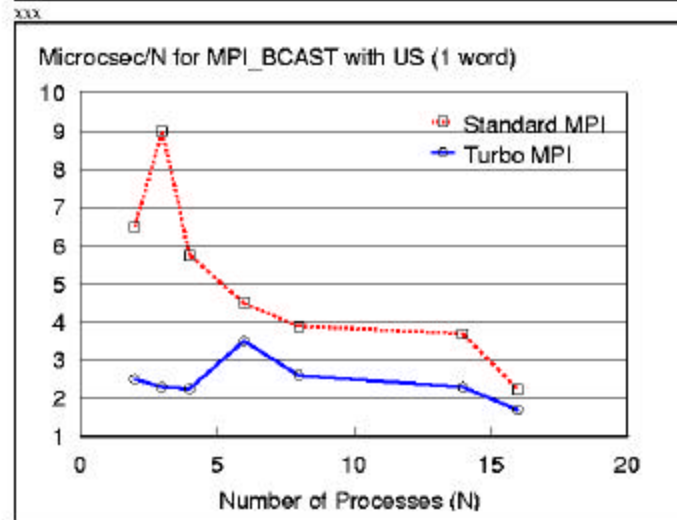
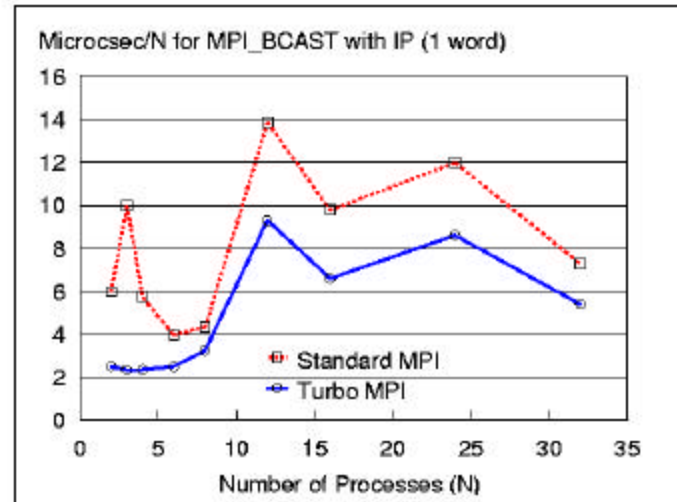
MPI_BCAST

The time taken for MPI_BCAST to broadcast one double precision word across all processors was measured for 2 to 32 processes on four 8-processor Power3 high nodes. The times are shown as Microsec/N (where N is the number of processes), simply to aid representation on a single chart.

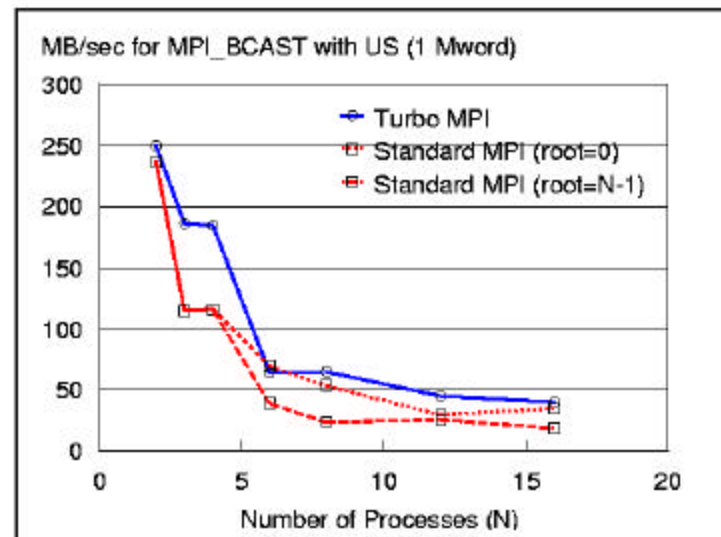
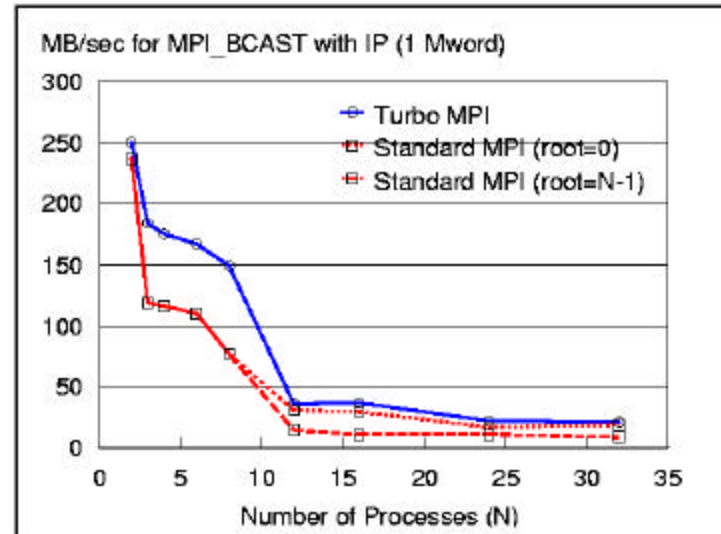
The IP communication was made using the SP switch connection between the nodes.

The broadcast rate for large arrays was measured in MB/sec for broadcasting an array of one million double precision words to all processors. For standard MPI, the rate depended on the value of the root processor.

MPI_BCAST



MPI_BCAST



MPI_ALLTOALL

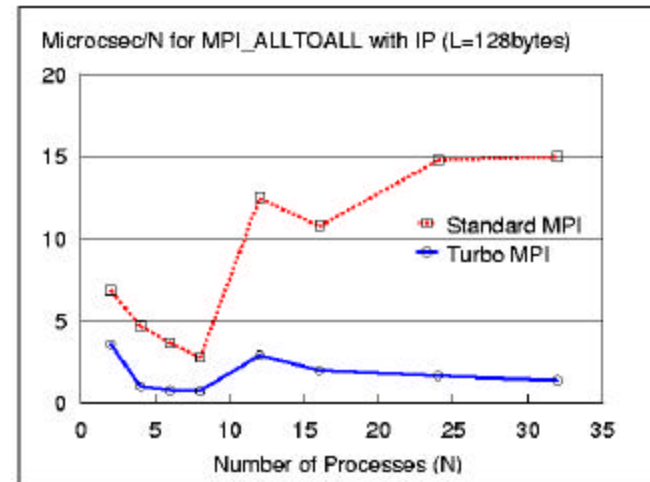
MPI_ALLTOALL

The time taken for MPI_ALLTOALL for each processor to communicate a minimal amount of data to each of N-1 other processors was measured for N = 2 to 32 on four 8-processor Power3 high nodes. The total amount of data transmitted by each node was approximately 128 bytes.

The times are shown as Microsec/N, simply to aid representation on a single chart.

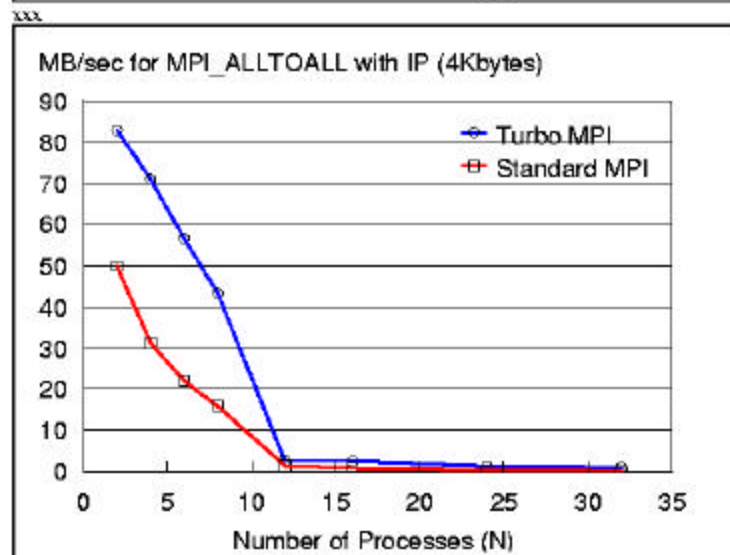
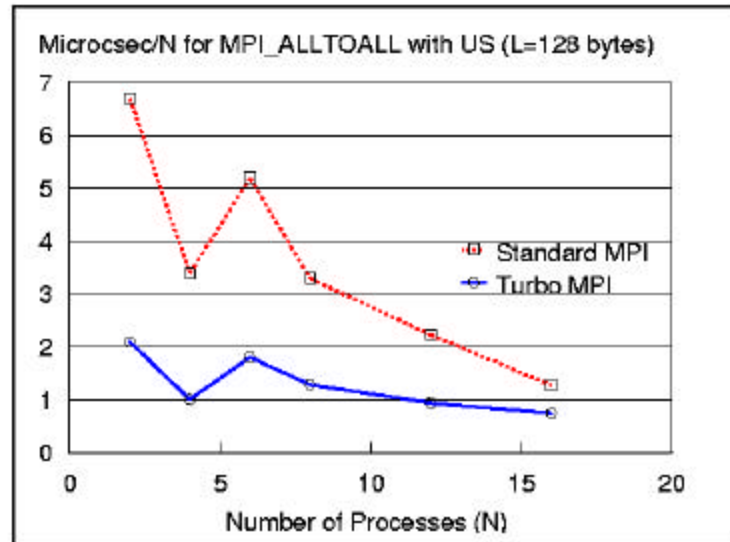
The IP communication was made using the SP switch connection between the nodes.

For large amounts of data, the communication time of Turbo MPI showed insignificant improvement over standard MPI. This is because it is not possible to reduce the total amount of data transmitted between nodes. The communication rate per processor is shown for the largest amount of data communication (4Kbytes total per processor) for which TurboMPI showed a significant improvement.



xxx

MPI_ALLTOALL



xxx

MPI_ALLTOALL

